

ToolMan: Upcoming Tools; Analyzing Paths

It's a new year, a new volume of *login*; perhaps time for new resolutions. A resolution I made late last year was to incorporate tools from other tool makers into ToolMan articles, primarily to keep the series more useful and interesting. Toward that goal I've included tools by two of my co-workers in the previous and current issues. I'll be writing about tools by people from the wider community in future articles, though I've found that working out the details in these situations takes much longer. But a few things are in the works, so please stay tuned.

Some topics I'm considering for future articles include:

email folder processing	RCS/SCCS wrappers
accounts	tar wrappers
netgroups	backups
disk quotas	directories/files
disk space	processes
finding	searching/replacing
comparing	sorting
text manipulation	printing
documenting	email alias parsing
remote execution	scheduling/calendar
Web (of course)	

In other words, the possibilities are wide open. Tools relating to these topics might be geared toward system administrators or general users. When possible, I'll survey several tools relating to a given topic. If you have any tools that fit this list or other categories that you would like me to include, please send a note.

Analyzing Paths

This issue I'll present a couple of tools for analyzing paths. One resolves symbolic links; the other shows status information for each component of a path (and follows links, too). Both are time savers in relevant situations.

Resolving Symbolic Links: `reslink`

As filesystems evolve and grow in complexity at your site, the tangle of symbolic links can become quite intractable. Sometimes you need to see where a path really goes, and how, in order to understand some situation or problem. Getting this information quickly, easily, and reliably would be nice.

Yuji Shinozaki, a fellow sysadmin here in my department, has written a tool named `reslink` for just such situations. It'll follow the links to a file (or some other filesystem object) and display various information about the links, depending on which options you choose.

For instance, at our site, `/usr/local/bin/` contains symbolic links to files actually residing in other filesystems. Sometimes these links, or various intermediate components, are themselves symbolic links and, well, you get the idea. It can become difficult to trace and grasp any particular one. `reslink` is an ideal tool for dissecting this maze of links.



by Daniel E. Singer

Dan has been doing a mix of programming and systems administration for 13 years. He is currently a systems administrator in the Duke University Department of Computer Science in Durham, North Carolina, USA.

<des@cs.duke.edu>

Tool: *reslink*

Abstract: *resolves symbolic links*

Platforms: *most UNIX*

Language: *Perl 5, plus standard modules: Getopt, Cwd*

Author: *Yuji Shinozaki*
<yuji@cs.duke.edu>

Availability:
<<http://www.cs.duke.edu/~yuji/tools/reslink>>
<<ftp://ftp.cs.duke.edu/pub/yuji/tools/reslink>>

`reslink` by default will list just the final path to the object specified as an argument. This can be useful in command substitution situations. For example:

```
% reslink /usr/local/bin/latex
/auto/pkg/tetex-0.4/tetex/bin/virtex
% ls -l `reslink /usr/local/bin/latex`
-rwxr-xr-x 1 lab lab 201548 Aug 7 1996 /auto/pkg\
/tetex-0.4/tetex/bin/virtex
```

With the `-t` (trace) and `-v` (verbose) options, details of all the links are shown:

```
% reslink -tv /usr/local/bin/latex
/usr/local/bin/latex => /auto/pkg/tetex-0.4/tetex/bin/virtex
/usr/local/bin/latex -> /auto/pkg/tetex-0.4/bin/latex
/auto/pkg/tetex-0.4/bin/latex -> /usr/pkg/tetex-0.4\
/tetex/bin/latex
/usr/pkg -> /auto/pkg
/auto/pkg/tetex-0.4/tetex/bin/latex -> virtex
```

Another handy option is the `-w` (which) option, which simulates the which command:

```
% reslink -wv latex
/usr/local/bin/latex => /auto/pkg/tetex-0.4/tetex/bin/virtex
% ls -l `reslink -w latex`
-rwxr-xr-x 1 lab lab 201548 Aug 7 1996 /auto/pkg/tetex-0.4\
/tetex/bin/virtex
```

There are a few other options (both real and planned), but I won't go into the details here. You can pick up a copy for yourself and play around with it.

The O'Reilly book *Programming Perl* by Larry Wall and Randal Schwartz includes a Perl script named `s1` that is similar to `reslink`, sans options. `s1` is available from your favorite CPAN site (paths vary) at `<ftp://.../scripts/nutshell/ch6/sl>`. It is also described in *UNIX Power Tools* by Jerry Peek, Tim O'Reilly, and Mike Loukides, and is available on the included CD archive and at `<ftp://ftp.ora.com/published/oreilly/nutshell/learning_perl/examples.tar.Z>`.

Seeing the Components of a Path: seepath

Sometimes you might need to see a little more about what's going on with a path. `seepath` is a tool for discovering problems with permissions and modes in a path by giving long-listing (`ls -l`) style status information on each of the path's components. For example, Shirley may tell you that she's running the new `sizzle` program and it's dying with the message `sizzle: cannot open /usr/project/sizzle/fizzle/drizzle`. To make a long story short, you could do the following:

```
% pwd
/usr/project
% seepath sizzle/fizzle/drizzle
seepath: /usr/project/sizzle/fizzle/drizzle
drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
drwxrwxr-x 35 root sys 1024 Nov 13 09:43 usr/
dr-xr-xr-x 2 root root 3 Dec 6 15:36 project/
drwxrwxr-x 34 ziggy eng 1024 Dec 2 15:53 sizzle/
drwxrwx--- 6 ziggy eng 512 Nov 20 17:46 fizzle/
-rw-rw-r- 1 ziggy eng 2674 Nov 5 19:14 drizzle
% groups shirley
user acctg mgmnt
```

The problem is now apparent: Shirley cannot access the `fizzle` directory.

seepath can also follow links and in fact will do just that when the `-F` (Follow) option is chosen. To use the example from the `reslink` discussion:

```
% seepath /usr/local/bin/latex
seepath: /usr/local/bin/latex
drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
drwxrwxr-x 35 root sys 1024 Nov 13 09:43 usr/
drwxrwxr-x 16 lab lab 512 Nov 13 09:43 local/
drwxrwxr-x 2 lab lab 16384 Nov 13 17:30 bin/
lrwxrwxrwx 1 root lab 29 Nov 13 09:41 latex -> \
/auto/pkg/tetex-0.4/bin/latex*

%seepath -F /usr/local/bin/latex
seepath: /usr/local/bin/latex
drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
drwxrwxr-x 35 root sys 1024 Nov 13 09:43 usr/
drwxrwxr-x 16 lab lab 512 Nov 13 09:43 local/
drwxrwxr-x 2 lab lab 16384 Nov 13 17:30 bin/
lrwxrwxrwx 1 root lab 29 Nov 13 09:41 latex -> \
/auto/pkg/tetex-0.4/bin/latex*

drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
dr-xr-xr-x 2 root root 6 Dec 6 16:05 auto/
drwxrwxr-x 376 lab lab 16384 Dec 3 12:06 pkg/
drwxr-xr-x 6 lab lab 8192 May 14 1997 tetex-0.4/
drwxr-xr-x 2 lab lab 8192 Nov 14 18:04 bin/
lrwxrwxrwx 1 lab lab 34 May 14 1997 latex -> \
/usr/pkg/tetex-0.4/tetex/bin/latex*

drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
drwxrwxr-x 35 root sys 1024 Nov 13 09:43 usr/
lrwxrwxrwx 1 root root 9 Aug 6 08:12 pkg -> /auto/pkg/
drwxr-xr-x 27 root root 1024 Sep 29 16:21 /
dr-xr-xr-x 2 root root 6 Dec 6 16:06 auto/
drwxrwxr-x 376 lab lab 16384 Dec 3 12:06 pkg/
drwxr-xr-x 6 lab lab 8192 May 14 1997 tetex-0.4/
drwxr-xr-x 5 ab lab 8192 Jun 18 10:13 tetex/
drwxr-xr-x 2 lab lab 8192 Nov 14 18:06 bin/
lrwxrwxrwx 1 lab lab 6 May 14 1997 latex -> virtex*
-rwxr-xr-x 1 lab lab 201548 Aug 7 1996 virtex*
```

Time for a little filesystem reorganizing, eh? With no arguments, `seepath` defaults to analyzing the path to the current directory.

Tools You Can Use

`reslink` and `seepath` are tools you can use to help diagnose problems with paths, in regards to both symbolic links and permissions/modes. Because they can also dramatically reduce the number of commands you need to type in certain situations, they might even help delay the onset of carpal tunnel syndrome! (Sorry, this does not constitute a warranty.)

If you see any ways in which these tools can be improved, the authors invite your comments. And remember, if you have tools that are worth sharing, please be sure to contact ToolMan.

Tool: `seepath`

Abstract: `displays status of all components of a path`

Platforms: `most UNIX`

Language: `Bourne shell`

Author: `Daniel E. Singer`
`<des@cs.duke.edu>`

Availability:
`<http://www.cs.duke.edu/~des/seepath.html>`
`<ftp://ftp.cs.duke.edu/pub/des/scripts/>`